REMARKS

In response to the Office Action dated June 1, 2004, Applicant respectfully requests reconsideration and withdrawal of the rejections of the claims and objections to the disclosure.

The Examiner is thanked for pointing out that a certified copy of the priority document has not yet been filed. Accordingly, a certified copy is being submitted herewith.

In response to the objection to the abstract, an amended abstract is being presented herein, in which the terms "PCSC" and "EMV" are explicitly defined.

The Office Action contains an objection to the drawings and a rejection of the claims under the first and second paragraphs of 35 U.S.C. § 112, stating that cases 1 and 3 of the ISO 7816-4 protocol are missing. The present invention is directed to the emulation of a smart card reader that conforms to the PSCS standard, to enable a personal computer to communicate with a reader that operates according to the EMV standard. The protocol for communicating with a smart card is defined in ISO 7816-4. Basically, communications are carried out by means of command-response pairs, in which each individual application protocol data unit (APDU) contains either a command message or a response message. The ISO specification defines four cases of command-response pairs. In two of these cases, namely cases 1 and 3, no data is expected in the response message. Conversely, in cases 2 and 4, data is expected. For the Examiner's reference, a copy of the relevant page of the ISO specification is attached.

In the case of the present invention, emulation is only necessary for cases 2 and 4, since they are the only ones in which data is expected in the response APDU.

Since cases 1 and 3 do not contain data, no emulation is necessary. See the specification at page 5, lines 21-24. In operation, when an APDU exchange falling into either case 1 or case 3 occurs, no action is taken by the emulator of the present invention. Consequently, these two cases are covered by the "no" response to decision block 14 in the figure, in which the response APDU is sent at step 20 without any intervening action. Accordingly, cases 1 and 3 are not affected by the objectives of the present invention, since they do not require emulation. As such, it is respectfully submitted that it is not necessary to describe them in the specification, or in the claims, to meet the requirements of 35 U.S.C. § 112.

The rejection under the first paragraph of 35 U.S.C. § 112 also states that the application does not define what cases 2 and 4 are. As noted above, these cases are defined in ISO 7816-4, as set forth in the specification, for example at page 3, lines 1-6. It is to be noted that patent specifications are written in accordance with the ordinary level of skill in the art to which they pertain. See MPEP 2163.01. In the present case, a person having an ordinary level of skill in the field of smart cards would be knowledgeable about the governing standards set forth in ISO 7816, and therefore be familiar with the various cases pertaining to APDU message structure. As such, it is not necessary to describe these cases in detail within the specification. Reconsideration and withdrawal of the rejections are respectfully requested.

Claims 1-4 were rejected under 35 U.S.C. § 103, on the grounds that they were considered to be unpatentable over the Levie et al. patent in view of the Bakker publication. It is respectfully submitted that neither of these references contains any disclosure relating to the emulation of a smart card reader that operates according to the PSCS standard, and therefore they cannot be interpreted to suggest the claimed subject matter, whether considered individually or in combination.

The Office Action states that the Levie patent teaches a modular terminal apparatus that includes an emulation interface, with reference to column 31, lines 37-43. This portion of the patent describes a Multiple Emulation PIN Pad Application (MEPPA) interface, that allows the point-of-sale terminal PIN pad to connect to various existing terminals. There is nothing in this portion of the patent, or elsewhere, however, which talks about emulation of a PCSC compatible reader.

Similarly, the Bakker article does not contain any such disclosure. The Office Action refers to its discussion of emulation at page 10, section 3.3. However, this portion of the article does not relate to the emulation of a PCSC-compatible reader. Rather, it describes the software emulation of a *smart card.* There is no discussion in the article relating to PCSC readers, let alone the emulation of such to permit communication with an EMV-based reader.

For at least this reason, therefore, the references cannot be deemed to suggest the claimed subject matter.

Furthermore, it is not apparent from the Office Action how the references are being interpreted relative to the claim language. MPEP § 2143 sets forth three basic requirements for a *prima facie* case of obviousness. The third requirement is that "the prior art reference (or references when combined) must teach or suggest all of the claim limitations." The Office Action does not identify where any of the steps recited in the claims are taught by the references. For instance, the first step of claim 1 is "determining the types of APDU exchanges for which emulation is to be effected." The Office Action has not pointed to any disclosure in either the Levie patent or the Bakker article which suggests such a step. On page 6, the Office Action merely reiterates the recitations of the claim, and then identifies certain passages from the Levie patent and the Bakker article. However, Applicant is unable to find any disclosures of the claimed step in these passages.

The second step recited in claim 1 is that of "emulating the return of a state word in compliance with the standards of the PCSC environment." Again, it is not apparent where this step is suggested by either of the references.

Similar considerations apply to the last three steps recited in claim 1, as well as the dependent claims. If the rejection based upon the Levie and Bakker references is not withdrawn, the Examiner is requested to identify, with particularity, where each claimed step can be found in the teachings of the references, or otherwise explain how the references are being interpreted in light of each of the claimed limitations.

In view of the foregoing, it is respectfully submitted that the pending claims are patentable over the applied references, and conform with the requirements of 35 U.S.C. § 112. Reconsideration and withdrawal of the rejections are respectfully requested.

Respectfully submitted,

BURNS, DOANE, SWECKER & MATHIS, L.L.P.

Date: August 31, 2004

By: James A. LaBarre
James A. LaBarre
Registration No. 28,632

P.O. Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

5.2.2 Security attributes

The security attributes, when they exist, define the allowed actions and the procedures to be performed to complete such actions.

Security attributes may be associated with each file and fix the security conditions that shall be satisfied to allow operations on the file. The security attributes of a file depend on

- — its category (DF or EF),
- — optional parameters in its file control information and/or in that of its parent file(s).

NOTE — Security attributes may also be associated to other objects (e.g., keys).

5.2.3 Security mechanisms

This part of ISO/IEC 7816 defines the following security mechanisms.

- — **Entity authentication with password** — The card compares data received from the outside world with secret internal data. This mechanism may be used for protecting the rights of the user.
- — **Entity authentication with key** — The entity to be authenticated has to prove the knowledge of the relevant key in an authentication procedure (e.g., using a GET CHALLENGE command followed by an EXTERNAL AUTHENTICATE command).
- — **Data authentication** — Using internal data, either secret or public, the card checks redundant data received from the outside world. Alternately, using secret internal data, the card computes a data element (cryptographic checksum or digital signature) and inserts it in the data sent to the outside world. This mechanism may be used for protecting the rights of a provider.
- — **Data encipherment** — Using secret internal data, the card deciphers a cryptogram received in a data field. Alternately, using internal data, either secret or public, the card computes a cryptogram and inserts it in a data field, possibly together with other data. This mechanism may be used to provide a confidentiality service, e.g., for key management and conditional access. In addition to the cryptogram mechanism, data confidentiality can be achieved by data concealment. In this case, the card computes a string of concealing bytes and adds it by exclusive-or to data bytes received from or sent to the outside world. This mechanism may be used for protecting privacy and for reducing the possibilities of message filtering.

The result of an authentication may be logged in an internal EF according to the requirements of the application.

5.3 APDU message structure

A step in an application protocol consists of sending a command, processing it in the receiving entity and sending back the response. Therefore a specific response corresponds to a specific command, referred to as a command-response pair.

An application protocol data unit (APDU) contains either a command message or a response message, sent from the interface device to the card or conversely.

In a command-response pair, the command message and the response message may contain data, thus inducing four cases which are summarized by table 4.

Table 4 — Data within a command-response pair

Case	Command data	Expected response data
1	No data	No data
2	No data	Data
3	Data	No data
4	Data	Data

5.3.1 Command APDU

Illustrated by figure 3 (see also table 6), the command APDU defined in this part of ISO/IEC 7816 consists of

- — a mandatory header of 4 bytes (CLA INS P1 P2),
- — a conditional body of variable length.

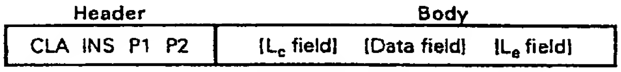

Figure 3 — Command APDU structure

The number of bytes present in the data field of the command APDU is denoted by L_c.

The maximum number of bytes expected in the data field of the response APDU is denoted by L_e (length of expected data). When the L_e field contains only zeroes, the maximum number of available data bytes is requested.

Figure 4 shows the 4 structures of command APDUs according to the 4 cases defined in table 4.

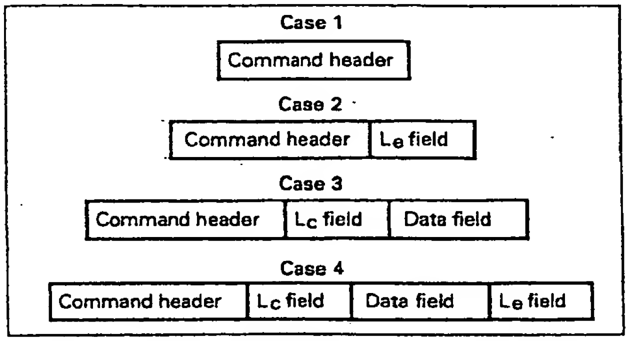

Figure 4 — The 4 structures of command APDUs